

MULTIMEDIA



UNIVERSITY

STUDENT ID NO

--	--	--	--	--	--	--	--	--	--

TABLE NO:

--

MULTIMEDIA UNIVERSITY

FINAL EXAMINATION

TRIMESTER 2, 2015/2016

TSN1101 – COMPUTER ARCHITECTURE AND ORGANIZATION

(All Sections /Groups)

08 MARCH 2016

9.00 a.m. - 11.00 a.m.

(2 Hours)

INSTRUCTIONS TO STUDENTS

1. This Question paper consists of **TWENTY NINE** pages (**excluding this page**) with **TWO SECTIONS**. Each section contains **FOUR** questions.
2. Attempt a total of **FIVE** questions, with a maximum of **THREE** questions from any one **SECTION** and the remaining **TWO** questions from the other **SECTION**. Each question carries **12 marks** and the distribution of the marks for each subdivision is given. Maximum allotted marks are **60 marks**.
3. Please **write all your answers in the Question Paper itself** in the space provided.

SECTION A**QUESTION A1:**

a) Perform the following conversions. Show the intermediate steps involved

(i) $(6CD.B)_{16} = (?)_8$

(ii) $(1001\ 0110\ 0011)_{8421\ code} = (?)_{Gray\ code}$

(2+2=4 marks)

Solution:

Continued.....

- b) Perform $(-57)_{10} - (71)_{10}$ by representing the numbers using **8-bit twos complement**.
State whether the **overflow flag** will be after performing the arithmetic operation.

(2 marks)

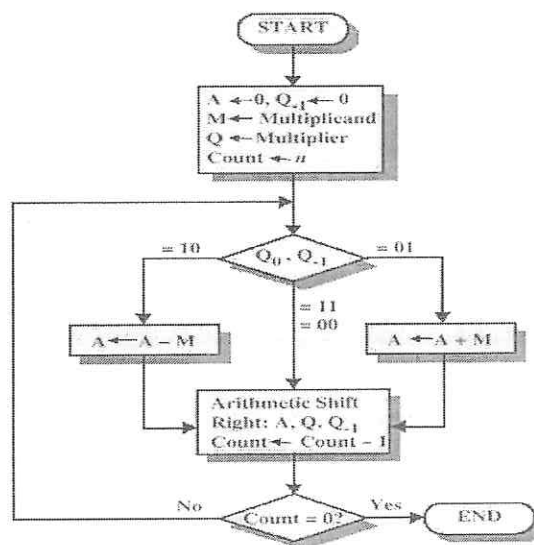
Solution:

- c) Show the steps using **Booth's algorithm** (Flowchart is given) for performing the **multiplication** of two 4-bit two's complement numbers as given below:

Multiplicand (M) = $(1010)_2$ or $(-6)_{10}$

Multiplier (Q) = $(1001)_2$ or $(-7)_{10}$

(4 marks)



Continued.....

Solution: $(-M) =$

A	Q	Q ₋₁	M	Operation	Count (n)	

Product = $AQ =$

Continued.....

- d) Represent the number $(360.5)_{10}$ or $(101101000.1)_2$ by using IEEE 32-bit floating-point format as given below.

Sign	Biased Exponent	Normalized Mantissa
(1 bit)	(8 bits)	(23 bits)

Perform the following steps:

- Normalize the given binary number into the form, $(1.bbbbbbb \times 2^n)$.
 - Identify the bias used and express the biased exponent in binary.
 - Pack the result into the IEEE 32-bit floating-point format given above.
- (0.5+1+0.5=2 marks)

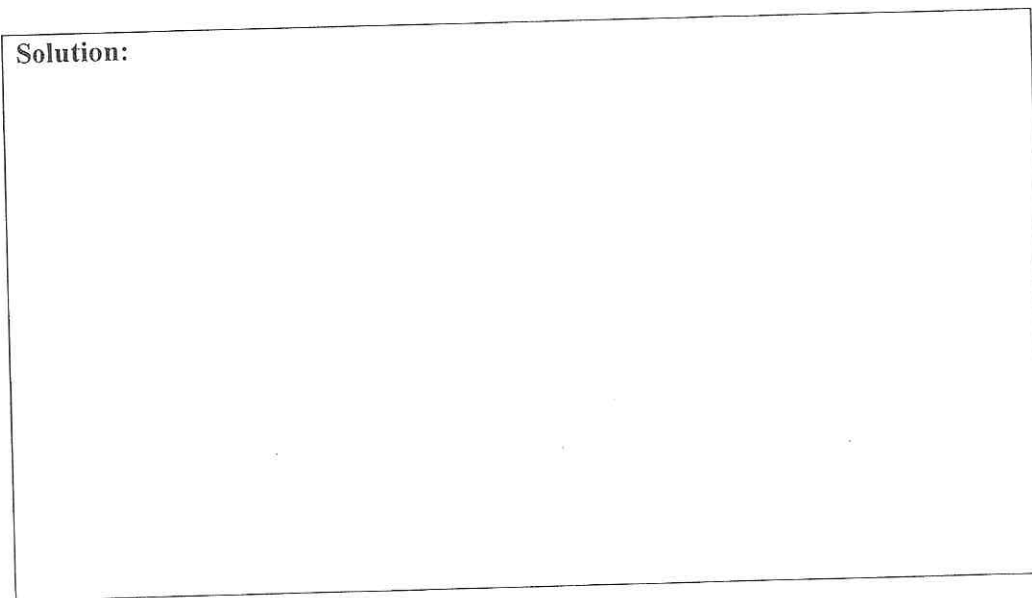
Solution:

Continued.....

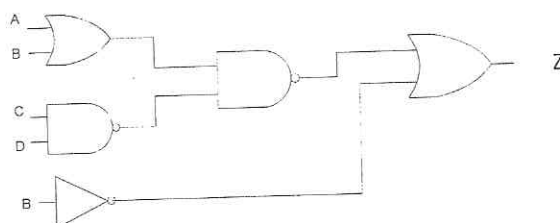
QUESTION A2:

- a) Prove the following equality by constructing the corresponding **truth tables** of the two-input logic gates.

$$f_{XNOR} (f_{NAND} (A, B), f_{NOR} (A, B)) = f_{XNOR} (A, B)$$

(2 marks)**Solution:**

- b) (i) Identify the original **output expression Z** for the circuit given below with inputs A, B, C and D.

**(1 mark)**

- (ii) **Simplify the Boolean expression** for the output Z by applying Boolean rules/laws and theorems. **(2 marks)**

- (iii) Draw the **logic diagram** for the simplified expression obtained in (ii) for the output Z. **(1 mark)**

Continued.....

Solution:

Continued.....

c) For the Boolean function, $F = (A\bar{B}) + (B\bar{C}) + (\bar{A}B\bar{C})$

- (i) Construct the appropriate **truth table**, and
- (ii) Find the **standard SOP** and **standard POS** expressions.

(2+2 =4 marks)

Solution:

(i)

INPUTS			OUTPUT
A	B	C	F
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

(ii)

Continued.....

- d) Boolean expression for 2-input XOR gate is $F = \bar{A} B + A \bar{B}$
Implement 2-input 'XOR gate' using only 'NAND' gates.

(2 marks)

Solution:

QUESTION A3:

- a) In a home security system, **three sensors (A1, A2 and A3)** are placed at different places to detect an intrusion. If two or more sensors are triggered, an intrusion is detected and an alarm will be activated.
Input signal lines indicate the sensors (if triggered, it is indicated as binary 1, if not triggered, it is indicated as binary 0) and the output signal line (R) indicates the detected (1) or undetected (0) intrusion and the corresponding action system.

- (i) Draw the **truth table** for the above problem indicating all possible combinations for the sensors and the corresponding status of intrusion detection and action system.
- (ii) Minimize the expression into **simplified SOP form using K-Map**.
- (iii) Construct a logic diagram using **AND-OR gate network**.

(2+1+1=4 marks)

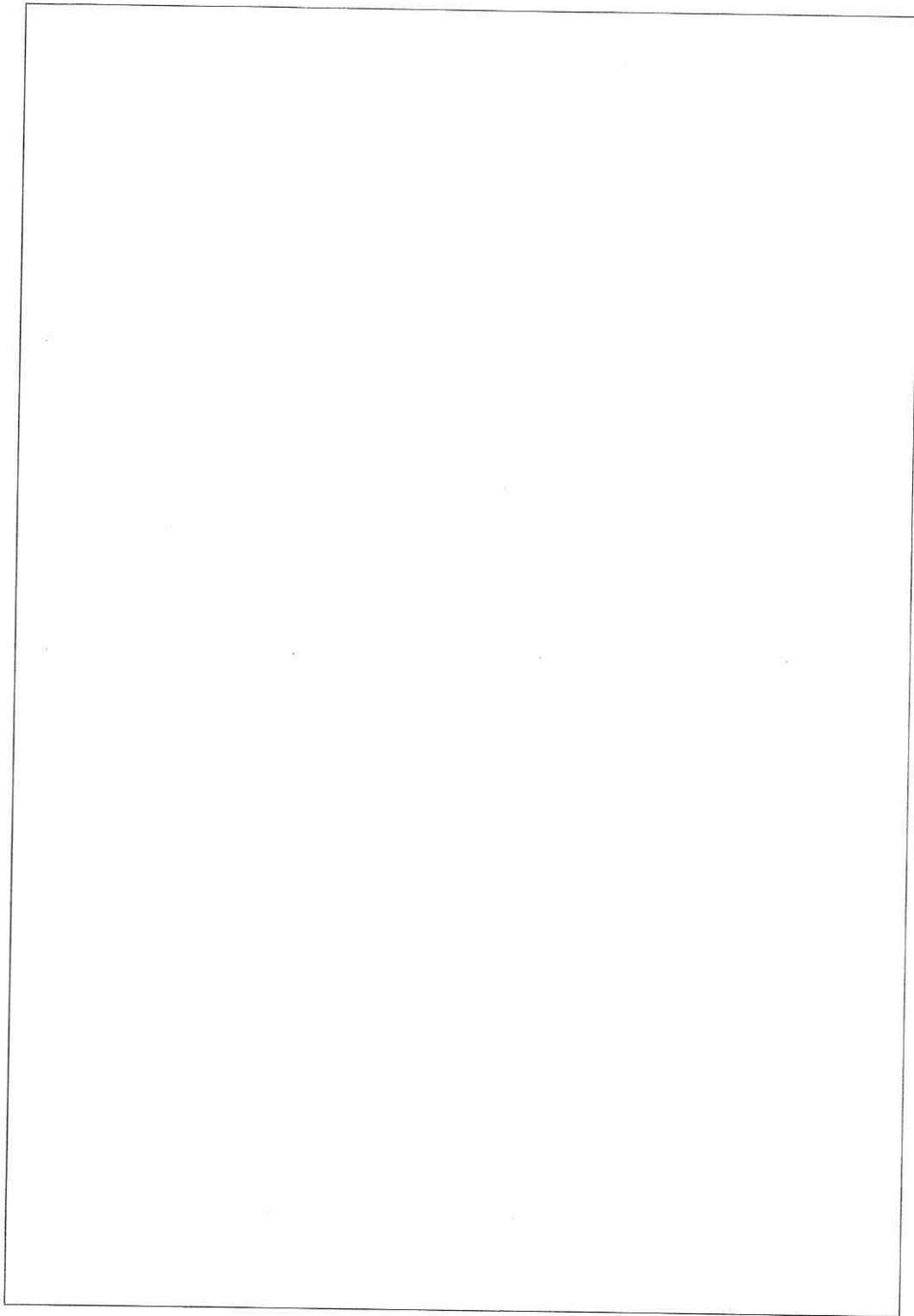
Continued.....

Solution:

(i) Truth Table:

INPUTS (Sensors)			OUTPUT (Intrusion Detection)
A1	A2	A3	R
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

Continued.....



Continued.....

- b) Implement the following Boolean expression, given as sum of minterms, using 8×1 multiplexer.

$$F(A, B, C, D) = \sum m(4, 6, 7, 9, 12, 13)$$

For implementation, perform the following steps:

- (i) Construct the **truth table** and evaluate F as 0, 1, variable D or the complement of variable D. (2 marks)
- (ii) Draw the **connection diagram** using 8×1 multiplexer. (2 marks)

Solution:

i) Truth Table:

A	B	C	D	F
0	0	0	0	
0	0	0	1	$F=$
0	0	1	0	
0	0	1	1	$F=$
0	1	0	0	
0	1	0	1	$F=$
0	1	1	0	
0	1	1	1	$F=$
1	0	0	0	
1	0	0	1	$F=$
1	0	1	0	
1	0	1	1	$F=$
1	1	0	0	
1	1	0	1	$F=$
1	1	1	0	
1	1	1	1	$F=$

Continued.....



Continued.....

c) Design a **2-bit binary-to-gray code converter** circuit by

- (i) Constructing the **truth table**,
- (ii) Identifying the **simplified Boolean expressions** for the gray code output, and
- (iii) Drawing the **simplified logic diagram**.

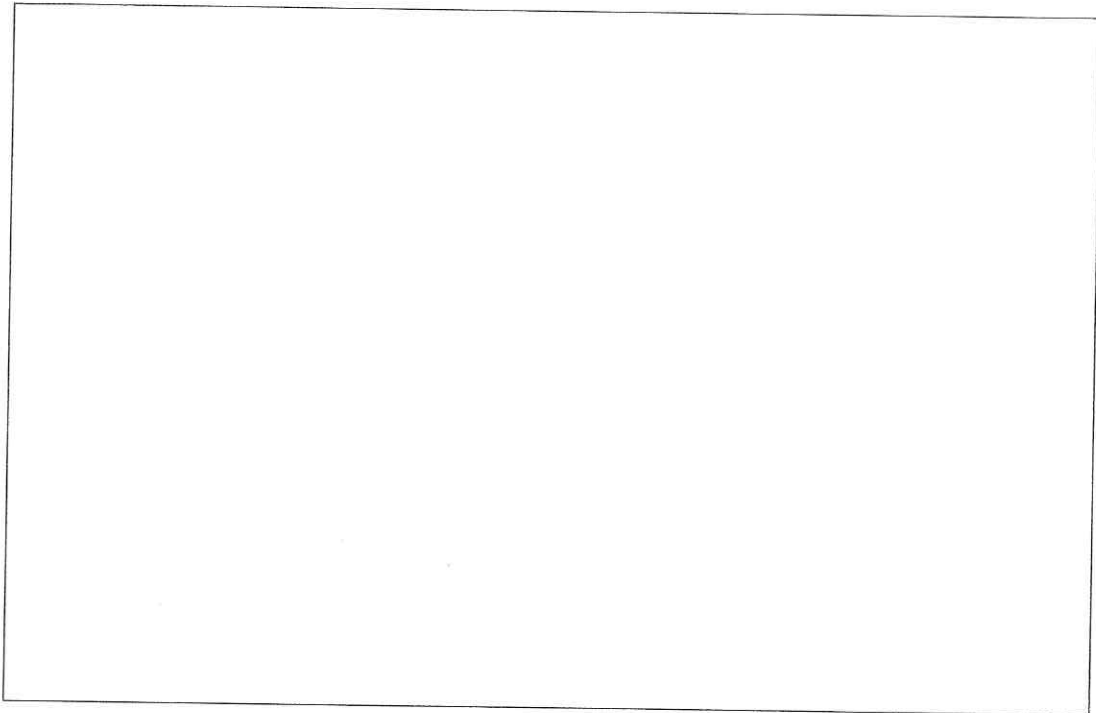
(1+2+1=4 marks)

Solution:

(i) Construction of Truth Table for 2-bit Binary-to-Gray code converter circuit:

Input (Binary)		Output (Gray code)	
B ₁	B ₀	G ₁	G ₀
0	0		
0	1		
1	0		
1	1		

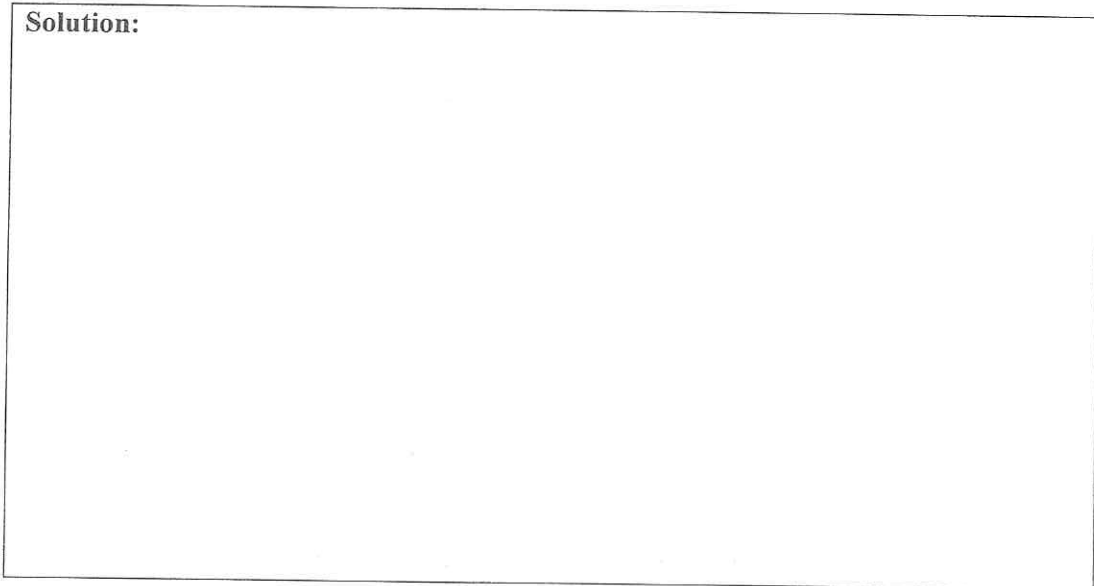
Continued.....

**QUESTION A4:**

- a) Draw the logic diagram of **Set-Reset Latch** using only **NOR gates**.

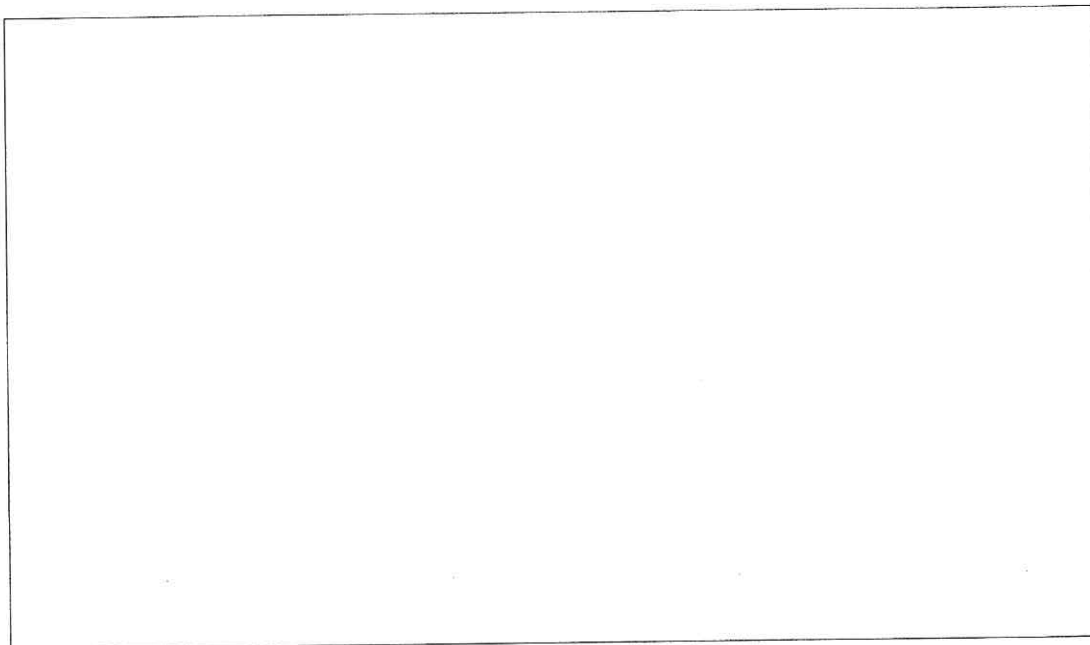
(2 marks)

Solution:



Continued.....

- b) Draw the logic diagram of a **3-bit binary ripple UP counter** using **J-K flip-flops** that trigger on the **positive-edge of the clock**. (2 marks)



- c) Design a **3-bit synchronous counter** using **J-K flip-flops** that produce the following sequence: **000, 001, 010, 011, 100, and repeat**. The undesired states must always go to 000 on the NEXT clock pulse. Your design should include
- (i) **State transition diagram** showing all possible states, including those that are not part of the desired counting sequence. (1 mark)
 - (ii) **Circuit excitation table** for $J_A, J_B, J_C, K_A, K_B, K_C$ using Present States and Next States (2 marks)
 - (iii) **Karnaugh Map simplification** for each of $J_A, J_B, J_C, K_A, K_B, K_C$ inputs. (3 marks)

Continued.....

Solution:

(ii) Circuit Excitation Table:

PRESENT State			NEXT State			J _C	K _C	J _B	K _B	J _A	K _A
C	B	A	C	B	A						
0	0	0									
0	0	1									
0	1	0									
0	1	1									
1	0	0									
1	0	1									
1	1	0									
1	1	1									

Continued.....

(iii) Karnaugh Map simplifications:

Continued.....

- d) Draw the **timing diagram** for a **4-bit serial in serial out shift register** constructed with positive edge triggered D flip-flops. Assume the data input **1001** and the register is initially cleared. (2 marks)

Solution:

Continued.....

SECTION B**QUESTION B1**

- a) Name **FOUR** major components of CPU and explain the function of each component.
(2 marks)

Solution:

- b) The basic instruction cycle in a computer consists of two steps, fetch and execute. State the **FOUR** steps involved in the fetch cycle.
(2 marks)

Solution:

Continued.....

- c) State any **FOUR** categories of instruction set from the aspect of functionality. List any **TWO** operations from each category.

(4 marks)

Solution:

- d) One of the issues in designing registers for microprocessor is whether to use completely general purpose registers or specialize the use of registers. Discuss the consequences (advantages and disadvantages) of the issue.

(4 marks)

Solution:

Continued.....

QUESTION B2

- a) After performing addition of two unsigned operands, 56_{10} and 11_{10} , on a computer with an 8-bit word, list the value of the following flags:
- i) Carry
 - ii) Zero
 - iii) Overflow
 - iv) Sign
 - v) Even parity
 - vi) Auxiliary carry

(3 marks)

Solution:

- b) State **THREE** types of addressing mode that do not need memory references. List an instruction from each category.

(3 marks)

Solution:

Continued.....

- c) Assume that Address 33 contains the value 55, Address 55 contains the value 66, Address 66 contains the value 77, and Address 77 contains the value 88. Given the memory values above and a one-address machine with an Accumulator (Register A), what values do the following instructions load into the Accumulator?
- | | |
|-----------------------|------------------------|
| i) LOAD IMMEDIATE 55 | (ii) LOAD DIRECT 55 |
| iii) LOAD INDIRECT 55 | (iv) LOAD IMMEDIATE 33 |
| v) LOAD DIRECT 77 | (vi) LOAD INDIRECT 66 |

(3 marks)

Solution:

- d) Describe and state the difference between LDAX and the LDA instruction of the 8085 microprocessor.

(3 marks)

Solution:

Continued.....

QUESTION B3

- a) List the micro-operations the micro-processor control unit must perform to add the content of memory location X to Register R1 (ADD R1, X).

(3 marks)

Solution:

- b) An external device attaches to the computer by a link to an I/O module. What are the **FIVE** major functions of an I/O module?

(2 marks)

Solution:

Continued.....

- a) Suppose that an 8-bit data word is stored in memory has the value 01101111. Using the Hamming Code algorithm, determine check bits (C1, C2, C4 and C8) that would be stored in memory with the data word, and complete the table below. Show the steps in calculating each check bit.

(5 marks)

Solution:

Bit Position	12	11	10	9	8	7	6	5	4	3	2	1
Bits	D8	D7	D6	D5	C8	D4	D3	D2	C4	D1	C2	C1
Word												
Check bit												

- c) In a symmetric multiprocessor (SMP) organization, multiple processors share a single memory space or pool of memory by means of a shared bus or other interconnection mechanism. What are the **FIVE** characteristics of a SMP computer?

(2 marks)

Solution:

Continued.....

QUESTION B4

- a) Specify the flag status of the following 8085 Assembly Language Program (ALP) instructions are executed:

ANI 00H
MVI B, FFH
INR A
ADD B

(2 marks)

Solution:

1. ANI 00H

S	Z	AC	P	CY

2. MVI B, FFH

S	Z	AC	P	CY

3. INR A

S	Z	AC	P	CY

4. ADD B

S	Z	AC	P	CY

Continued.....

- b) Given the following register and memory values for an Intel 8085 microprocessor, what values are in the Accumulator (Register A) after these instructions are executed? (Assume that each instruction is independently executed)

Memory addresses:

- Register A contains 30H
- Register B contains 70H
- Register C contains 05H
- Register H contains 60H
- Register L contains 20H

4050H	60H
5001H	40H
6020H	02H
7005H	15H

- i. MVID, 20H
- ii. LDAX B
- iii. MOV A, M
- iv. MVI A, 30H
- v. MOV A, B
- vi. MOV M, A
- vii. ADD M
- viii. ADD C

(4 marks)

Solution:

Continued.....

- c) 8 numbers are stored in memory locations from 8001H to 8008H. Write an 8085 ALP to increment the numbers in these memory locations by 2.

(3 marks)

Solution:

- d) Write an 8085 ALP to:

- Store the values '20H' and 'FFH' in the memory locations 5000H and 6000H respectively
- Find the 2's complement of the sum of these two numbers and store the result in memory location 7000H

(3 marks)

Solution:

Continued.....

Appendix

Intel 8085 Instruction Set

Hex	Mnemonic	Hex	Mnemonic	Hex	Mnemonic	Hex	Mnemonic
CE	ACI 8-Bit	2B	DCX H	52	MOV D,D	E5	PUSH H
8F	AX A	3B	DCX SP	53	MOV D,E	F5	PUSH PSW
88	ADC B	F3	DI	54	MOV D,H	17	RAL
89	ADC C	FB	EI	55	MOV D,L	1F	RAR
8A	ADC D	76	HLT	56	MOV D,M	D8	RC
8B	ADC E	DB	IN 8-Bit	5F	MOV E,A	C9	RET
8C	ADC H	3C	INR A	58	MOV E,B	20	RIM
8D	ADC L	04	INR B	59	MOV E,C	07	RLC
8E	ADC M	0C	INR C	5A	MOV E,D	F8	RM
87	ADD A	14	INR D	5B	MOV E,E	D0	RNC
80	ADD B	1C	INR E	5C	MOV E,H	C0	RNZ
81	ADD C	24	INR H	5D	MOV E,L	F0	RP
82	ADD D	2C	INR L	5E	MOV E,M	E8	RPE
83	ADD E	34	INR M	67	MOV H,A	E0	RPO
84	ADD H	03	INX B	60	MOV H,B	0F	RRC
85	ADD L	13	INX D	61	MOV H,C	C7	RST 0
86	ADD M	23	INX H	62	MOV H,D	CF	RST 1
C6	ADI 8-Bit	33	INX SP	63	MOV H,E	D7	RST 2
A1	ANA A	DA	JC 16-Bit	64	MOV H,H	DF	RST 3
A0	ANA B	FA	JM 16-Bit	65	MOV H,L	E7	RST 4
A1	ANA C	C3	JMP 16-Bit	66	MOV H,M	EF	RST 5
A2	ANA D	D2	JNC 16-Bit	6F	MOV L,A	F7	RST 6
A3	ANA E	C2	JNZ 16-Bit	68	MOV L,B	FF	RST 7
A4	ANA H	F2	JP 16-Bit	69	MOV L,C	C8	RZ
A5	ANA L	EA	JPE 16-Bit	6A	MOV L,D	9F	SBB A
A6	ANA M	E2	JPO 16-Bit	6B	MOV L,E	98	SBB B
E6	AM 8-Bit	CA	JZ 16-Bit	6C	MOV L,H	99	SBB C
CD	CALL 16-Bit	3A	LDA 16-Bit	6D	MOV L,L	9A	SBB D
DC	CC 16-Bit	0A	LDAX B	6E	MOV L,M	9B	SBB E
FC	CM 16-Bit	1A	LDAX D	77	MOV M,A	9C	SBB H
2F	CMA	2A	LHLD 16-Bit	70	MOV M,B	9D	SBB L
3F	CMC	01	LXI B,16-Bit	71	MOV M,C	9E	SBB M
BF	CMP A	11	LXI D,16-Bit	72	MOV M,D	DE	SBI 8-Bit
B8	CMP B	21	LXI H,16-Bit	73	MOV M,E	22	SHLD, 16-Bit
B9	CMP C	31	LXI SP,16-Bit	74	MOV M,H	30	SIM
BA	CMP D	7F	MOV A,A	75	MOV M,L	F9	SPHL
BB	CMP E	78	MOV A,B	3E	MVI A,8 Bit	32	STA 16-Bit
BC	CMP H	79	MOV A,C	06	MVI B,8 Bit	02	STAX B
BD	CMP L	7A	MOV A,D	0E	MVI C,8 Bit	12	STAX D
BE	CMP M	7B	MOV A,E	16	MVI D,8 Bit	37	STC
D4	CNC 16-Bit	7C	MOV A,H	1E	MVI E,8 Bit	97	SUB A
C4	CNZ 16-Bit	7D	MOV A,L	26	MVI H,8 Bit	90	SUB B
F4	CF 16-Bit	7E	MOV A,M	2E	MVI L,8 Bit	91	SUB C
EC	CPE 16-Bit	47	MOV B,A	36	MVI M,8 Bit	92	SUB D
FE	CPI 8-Bit	40	MOV B,B	00	NOP	93	SUB E
E4	CPO 16-Bit	41	MOV B,C	B7	ORA A	94	SUB H
CC	CZ 16-Bit	42	MOV B,D	B0	ORA B	95	SUB L

Continued.....

27	DAA		43	MOV	B,E	B1	ORA	C	96	SUB	M
09	DAD	B	44	MOV	B,H	B2	ORA	D	D6	SUI	8-Bit
19	DAD	D	45	MOV	B,L	B3	ORA	E	EB	XCHG	
29	DAD	H	46	MOV	B,M	B4	ORA	H	AF	XRA	A
39	DAD	SP	47	MOV	C,A	B5	ORA	L	A8	XRA	B
3D	DCR	A	48	MOV	C,B	B6	ORA	M	A9	XRA	C
05	DCR	B	49	MOV	C,C	F6	ORI	8-Bit	AA	XRA	D
0D	DCR	C	4A	MOV	C,D	D3	OUT	8-Bit	AB	XRA	E
15	DCR	D	4B	MOV	C,E	E9	PCHL		AC	XRA	H
1D	DCR	E	4C	MOV	C,H	C1	POP	B	AD	XRA	L
25	DCR	H	4D	MOV	C,L	D1	POP	D	AE	XRA	M
2D	DCR	L	4E	MOV	C,M	E1	POP	H	EE	XRI	8-Bit
35	DCR	M	57	MOV	D,A	F1	POP	PSW	E3	XTHL	
0B	DCX	B	50	MOV	D,B	C5	PUSH	B			
1B	DCX	D	51	MOV	D,C	D5	PUSH	D			

End of Page